

第2回　1980年代後半，ネットリスト形式と動作記述言語が乱立

明石貴昭

Verilog HDLが登場した1980年代中ごろまで，論理回路設計では回路図エディタを使って論理ゲートのシンボルをつないでいく，いわゆるゲート・レベル設計が中心でした．このころ，二つの新しい動きが出てきました．一つは各EDAツール間のインターフェースをとろうという動き，もう一つは設計抽象度を引き上げて，設計の効率化を図ろうという動きです．

● 独自のネットリスト・フォーマットや記述言語が乱立

当時の設計のゴールデン・データ（リファレンスとなるもっとも重要なデータ）は回路図でした．ただし，回路図エディタを供給するEDAベンダ各社はそれぞれ独自のフォーマットを採用しており，専用のトランスレータを開発しない限り，ツール間のインターフェースがとれない状況でした．

この問題を解消するため，米国EIA（Electronic Industries Association）はEDIF（イーディフ）（electronic design interchange format）というネットリストの共通フォーマットを規格化しました．EDIFは回路図の座標情報が取り扱えるなど，回路図データ（ゲート・レベル・ネットリスト）の標準的な交換フォーマットとして大きな期待が寄せられていました．しかし，残念ながらEDIFは当時のEDAベンダごとに方言が多数存在し，それが汎用性を低下させ，あまり成功しませんでした．論理シミュレータも開発元ごとに異なるデータ・フォーマットを採用しており，1度ユーザがツールを選定すると，いつまでもそのベンダについていかざるをえないという状況が続いていました．

もう一つの流れである設計抽象度の引き上げについては，多数のベンダが競って「動作記述言語」を開発していました．当時はまだ，RTL（register transfer level）などの抽象度がきちんと定義されておらず，「動作（回路のふるまい）を表現する」という意味でこう呼ばれていました．多くのEDAベンダは，動作モデルをC言語ライクな独自言語で定義しようとしました．Verilog HDLも当初はこうした言語の一つでしかありませんでした．

このころのおもなシミュレータには，米国Daisy Systems社の「Logician」，英国GenRad社の「HILO 3」，米国HHB Systems社の「CADAT」，米国Simucad社の「SILOS II」，米国Tegas Systems社の「Tegas」，米国Teradyne社の「LASAR」，米国Viewlogic Systems社の「ViewSim」などがあります．HILO 3を出荷していたGenRad社には，Peter L. Flake氏（現在，米国Synopsys社Scientist）とPhilip R. Moorby氏の姿がありました．両氏は後に米国Co-Design Automation社（Synopsys社が吸収合併）に参加し，SystemVerilogの設計記述の基礎となったSuperlog言語やその処理系の開発に携わっています．

● Verilog-XLが優れていたからVerilog HDLは生き残れた

1980年代後半に生まれた多くの動作記述言語のうち，なぜVerilog HDLだけが生き残ったのでしょう？ 今から考えると，以下のような理由が思い浮かぶと思います．

a）完成度の高い優れた言語だった

b）設計抽象度をRTLに引き上げたかった（当時，Synopsys社の論理合成ツール「Design Compiler」が登場していた）

c）シミュレータ「Verilog-XL」の完成度が高かった

ほとんどの方は，**b）**がおもな理由であると予想するでしょう．しかし，筆者は**c）**の意見に近く，「Verilog-XLのゲート・レベル・シミュレーションが高速だったから」であると思っています．

前述したように，この時代はゲート・レベル設計が全盛で，増大する回路規模に対応するため，高速なシミュレータが求められていました．現在のネイティブ・コンパイル型シミュレータに慣れた方々から見ると，インタープリタ型のVerilog-XLはかなり遅く感じられるかもしれません．しかし当時は，「ソフトウェア・アクセラレータ」などという意味不明なキャッチで呼ばれるほど高速でした．競合シミュレータの速度を2けたは確実に上回っており，ベンチマークでは無敵だったのです．

Verilog HDLが使われ始めたおもな理由がシミュレーション性能であったにしろ，Verilog HDLは同時にさまざまなメリットを業界にもたらしました．その一つとして，ネットリストとシミュレーション・パターンの言語を統一したことが挙げられます．

Verilog-XL出現以前の各シミュレータの入力フォーマットは各社ばらばらでしたが，それらのフォーマットは記述言語ではなく，さらにネットリスト用フォーマットとシミュレーション・パターン用フォーマットは別々でした．一方，Verilog HDLは単一言語でネットリスト，動作記述，テストベンチ記述を表現でき，これにより「DUT（device under test）の周辺回路の動作記述」イコール「テストベンチ」という考えかたが広まりました．

この状況は，現在のSystemVerilogが示している「設計記述と検証記述の統一言語」というコンセプトに似ていると思いませんか？ やはり時代は少しだけ形を変えて繰り返すのでしょうか．

<筆者プロフィール>

明石 貴昭．ケイデンス・デザイン・システムズ入社後，Verilog-XL，NC-Simシミュレータなどを担当．現在，日本シノプシスにて論理検証プラットホームDiscoveryやSystemVerilog関連のしごとに従事している．旧EIAJ（現JEITA）1364-1995プロジェクトに参加し，現在はJEITA/EDA技術専門委員会SystemVerilogタスクグループの副主査として，SystemVerilogの標準化に携わっている．